DETAILED ACTION

1. This Office Action has been issued in response to Applicant's Amendment filed April 23, 2009.

2. Claims 31, 32, 37, 38, 43 and 44 have been amended. Claims 31-48 have been examined and are pending.

Priority

3. Applicant's claim for the benefit of a prior-filed application under 35 U.S.C. 119(e) or under 35 U.S.C. 120, 121, or 365(c) is acknowledged. Applicant has not complied with one or more conditions for receiving the benefit of an earlier filing date under 35 U.S.C. 119 as follows:

4. MPEP 201.11 (III)(D) states: If an applicant includes a benefit claim in the application but not in the manner specified by 37 CFR 1.78(a) (e.g., if the claim is included in an oath or declaration or the application transmittal letter) within the time period set forth in 37 CFR 1.78(a), the Office will not require a petition under 37 CFR 1.78(a) and the surcharge under 37 CFR 1.17(t) to correct the claim if the information concerning the claim was recognized by the Office as shown by its inclusion on the filing receipt. If, however, a claim is not included in the first sentence(s) of the specification or in an ADS and is not recognized by the Office as shown by its absence on the filing receipt, the Office will require a petition under 37 CFR 1.78(a) and the surcharge to correct the claim. The Office may not recognize any benefit claim where there is no indication of the relationship between the nonprovisional applications or no indication of the intermediate nonprovisional application that is directly claiming the benefit of the provisional application.

5. In this situation the provisional priority was claimed however there was no indication of the serial number of the provisional application until May 3, 2007, as such the office was unable to recognize a benefit claim within the time period set forth in 37 CFR 1.78(a)(2)(ii). For these reasons a petition should be filed to correct the claim.

Response to Arguments

6. Applicant's arguments filed April 23, 2009 have been fully considered but they are not persuasive.

7. Applicant's arguments with respect to claims 31-48 have been considered but are moot in view of the new ground(s) of rejection.

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

9. The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.

4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

10. This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

11. Claims 31-33, 36-39, 42-45 and 48 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Pat. No. 5892919 to Nielsen (hereinafter "Nielsen") and further in view of US Pat. No. 6092100 to Berstis et al. (hereinafter "Berstis").

12. **As to Claim 31,** Nielsen discloses **a system for translating domain names comprising:**

a Uniform Resource Locater (URL) detection module, configured to:

receive a URL request by a user to access a destination fully qualified domain name (FQDN) (Figure 4 of Nielsen discloses a user issuing a GET command for a network address such as a URL (400), then figure 5 discloses looking up the issued URL in the spell check cache (500). As such it is seen that because the invention looks up the issued URL in its spell check cache, that it must have received the issued URL)**, and**

a URL redirection module, configured to:

receive the invalid URL request from the URL detection module (Figure 5 of Nielsen discloses processing the requested URL to see if it can find the associated correct URL (515,520). This is seen to be part of the FQDN mapping module. Since the FQDN mapping module receives the requested URL for processing it is seen that another component must have redirected the URL to the FQDN mapping module. As such it is further seen that that component must have received the invalid URL request as well)**, and**

redirect the invalid URL request to a FQDN translation module (Figure 5 of Nielsen discloses processing the requested URL to see if it can find the associated correct URL (515,520). This is seen to be part of the FQDN mapping module. Since the FQDN mapping module receives the requested URL for processing it is seen that another component must have redirected the URL to the FQDN mapping module)**; and**

the FQDN translation module, configured to:

translate the invalid URL request to a target valid FQDN using a FQDN mapping module (Figure 5 of Nielsen discloses returning the correct URL from the originally invalid URL and then issuing that URL instead of the original URL (545, 550). Thus it is seen that the invalid URL has been translated to the correct URL)**, wherein the FQDN mapping module is stored on a computer readable storage medium** (Column 5 lines 10-20 of Nielsen disclose memory media will contain the program information for controlling the computer to enable the computer to perform its functions in accordance with the invention)**.**

Nielsen does not explicitly disclose **determine that the URL request is an invalid URL request when the URL request is inconsistent with a predefined URL stored in a cookie, wherein the predefined URL stored in the cookie is specified by the user;**

However, Berstis discloses this (Column 5 line 50 – Column 6 line 16 and figure 4 of Berstis disclose at step 52 a test is done to determine whether the string entered in the address field (URL) is recognized. An address is considered recognized if the client has made contact with that URL and thus the list of recognized URLs consists of any past URL the user has been able to access. If the string is recognized (valid) then the client is connected to the target URL, if the string is not recognized (inconsistent with predefined URL) then the process continues to try to correct the string. It is seen that the list of recognized URLs were specified by the user because it is a list of previously contacted URLs, and in order for the URLs to be previously contacted they must have originally been specified by the user in an attempt to access those URLs)

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the URL correction as disclosed by Nielsen, with detecting invalid URLs using a list of previously contacted URLs as disclosed by Berstis. One of ordinary skill in the art would have been motivated to combine to use a known technique to improve similar devices in the same way. Nielsen and Berstis are both directed toward identifying incorrect URLs and correcting them automatically for the user. As such it would be obvious to implement the features of either invention with each other to improve similar systems in the same way.

13. **As to Claim 32,** Nielsen-Berstis discloses the invention as claimed as described in claim 31, **further comprising:**

a FQDN default setter configured to provide a predefined default target valid FQDN, wherein the FQDN default setter is used by the FQDN mapping module (Figure 5 of Nielsen

discloses if the invention is unable to conclusively correct the invalid URL it will return a page to the user with the candidate URL and a request for other candidates. This is seen to be a default target valid FQDN, as it is the default if the correction to the invalid URL is not readily available. This page is seen to be predefined because it has been set by the invention to be the default page).

14. **As to Claim 33,** Nielsen-Berstis discloses the invention as claimed as described in claim 31**, wherein the FQDN mapping module is configured to provide a mapping between the invalid URL request and the target valid FQDN** (Figure 3 of Nielsen discloses a table that holds the invalid URLs and the correct URLs that they have been mapped to and then figure 5 discloses returning the correct URL from the originally invalid URL (545). This is seen to be having provided a mapping between the invalid URL and target valid FQDN).

15. **As to Claim 36,** Nielsen-Berstis discloses the invention as claimed as described in claim 31**, wherein the URL detection module, the URL redirection module, and the FQDN translation module execute in a browser** (Column 5 lines 20-25 of Nielsen disclose the user's computing device running a network browser such as a WWW browser software. Then column 2 lines 55-60 disclose the spell checking will transparently correct the URL and instruct the browser to return the document addressed by the corrected URL. Since the spell checker is able to instruct the browser it is seen to be executing inside the browser. As such it is seen that all associated modules are executing within the browser).

16. **As to Claim 37,** Nielsen discloses **a method for translating domain names, comprising:**

receiving, by a Uniform Resource Locater (URL) detection module, a URL request from a user to access a destination fully qualified domain name (FQDN) (Figure 4 of Nielsen discloses a user issuing a GET command for a network address such as a URL (400), then figure 5 discloses looking up the issued URL in the spell check cache (500). As such it is seen that because the invention looks up the issued URL in its spell check cache, that it must have received the issued URL)**, and**

receiving, by a URL redirection module, the invalid URL request from the URL detection module (Figure 5 of Nielsen discloses processing the requested URL to see if it can find the associated correct URL (515,520). This is seen to be part of the FQDN mapping module. Since the FQDN mapping module receives the requested URL for processing it is seen that another component must have redirected the URL to the FQDN mapping module. As such it is further seen that that component must have received the invalid URL request as well)**;**

redirecting, by the URL redirection module, the invalid URL request to a FQDN translation module (Figure 5 of Nielsen discloses processing the requested URL to see if it can find the associated correct URL (515,520). This is seen to be part of the FQDN mapping module. Since the FQDN mapping module receives the requested URL for processing it is seen that another component must have redirected the URL to the FQDN mapping module)**;**

translating, by the FQDN translation module, the invalid URL request to a target valid FQDN using a FQDN mapping module (Figure 5 of Nielsen discloses returning the correct

URL from the originally invalid URL and then issuing that URL instead of the original URL (545, 550). Thus it is seen that the invalid URL has been translated to the correct URL)**; and directing the user to a web site associated with the target valid FQDN** (Figure 5 of Nielsen discloses returning the correct URL from the originally invalid URL and then issuing that URL instead of the original URL (545, 550). Thus it is seen that the invalid URL has been translated to the correct URL, which was then issued)**.**

Nielsen does not explicitly disclose **determining, by the URL detection module that the URL request is an invalid URL request when the URL request is inconsistent with a predefined URL stored in a cookie, wherein the predefined URL stored in the cookie is specified by the user**

However, Berstis discloses this (Column 5 line 50 – Column 6 line 16 and figure 4 of Berstis disclose at step 52 a test is done to determine whether the string entered in the address field (URL) is recognized. An address is considered recognized if the client has made contact with that URL and thus the list of recognized URLs consists of any past URL the user has been able to access. If the string is recognized (valid) then the client is connected to the target URL, if the string is not recognized (inconsistent with predefined URL) then the process continues to try to correct the string. It is seen that the list of recognized URLs were specified by the user because it is a list of previously contacted URLs, and in order for the URLs to be previously contacted they must have originally been specified by the user in an attempt to access those URLs)

Examiner recites the same rationale to combine used in claim 31.

17. **As to Claim 38,** Nielsen-Berstis discloses the invention as claimed as described in claim 37**, further comprising:**

providing a predefined default target valid FQDN by a FQDN default setter, wherein the FQDN default setter is used by the FQDN mapping module (Figure 5 of Nielsen discloses if the invention is unable to conclusively correct the invalid URL it will return a page to the user with the candidate URL and a request for other candidates. This is seen to be a default target valid FQDN, as it is the default if the correction to the invalid URL is not readily available. This page is seen to be predefined because it has been set by the invention to be the default page)**.**

18. **As to Claim 39,** Nielsen-Berstis discloses the invention as claimed as described in claim 37**, wherein the FQDN mapping module is configured to provide a mapping between the invalid URL request and the target valid FQDN** (Figure 3 of Nielsen discloses a table that holds the invalid URLs and the correct URLs that they have been mapped to and then figure 5 discloses returning the correct URL from the originally invalid URL (545). This is seen to be having provided a mapping between the invalid URL and target valid FQDN)**.**

19. **As to Claim 42,** Nielsen-Berstis discloses the invention as claimed as described in claim 37**, wherein the URL detection module, the URL redirection module, and the FQDN translation module execute in a browser** (Column 5 lines 20-25 of Nielsen disclose the user's computing device running a network browser such as a WWW browser software. Then column 2 lines 55-60 disclose the spell checking will transparently correct the URL and instruct the browser to return the document addressed by the corrected URL. Since the spell checker is able

to instruct the browser it is seen to be executing inside the browser. As such it is seen that all associated modules are executing within the browser).

20. **As to Claim 43,** Nielsen discloses **a computer readable medium comprising executable instructions for translating domain names by:**

receiving, by a Uniform Resource Locater (URL) detection module, a URL request from a user to access a destination fully qualified domain name (FQDN) (Figure 4 of Nielsen discloses a user issuing a GET command for a network address such as a URL (400), then figure 5 discloses looking up the issued URL in the spell check cache (500). As such it is seen that because the invention looks up the issued URL in its spell check cache, that it must have received the issued URL)**, and**

receiving, by a URL redirection module, the invalid URL request from the URL detection module (Figure 5 of Nielsen discloses processing the requested URL to see if it can find the associated correct URL (515,520). This is seen to be part of the FQDN mapping module. Since the FQDN mapping module receives the requested URL for processing it is seen that another component must have redirected the URL to the FQDN mapping module. As such it is further seen that that component must have received the invalid URL request as well;

redirecting, by the URL redirection module, the invalid URL request to a FQDN translation module (Figure 5 of Nielsen discloses processing the requested URL to see if it can find the associated correct URL (515,520). This is seen to be part of the FQDN mapping module. Since the FQDN mapping module receives the requested URL for processing it is seen that another component must have redirected the URL to the FQDN mapping module);

translating, by the FQDN translation module, the invalid URL request to a target valid FQDN using a FQDN mapping module (Figure 5 of Nielsen discloses returning the correct URL from the originally invalid URL and then issuing that URL instead of the original URL (545, 550). Thus it is seen that the invalid URL has been translated to the correct URL)**; and directing the user to a web site associated with the target valid FQDN** (Figure 5 of Nielsen discloses returning the correct URL from the originally invalid URL and then issuing that URL instead of the original URL (545, 550). Thus it is seen that the invalid URL has been translated to the correct URL, which was then issued)**.**

Nielsen does not explicitly disclose **determining, by the URL detection module that the URL request is an invalid URL request when the URL request is inconsistent with a predefined URL stored in a cookie, wherein the predefined URL stored in the cookie is specified by the user**

However, Berstis discloses this (Column 5 line 50 – Column 6 line 16 and figure 4 of Berstis disclose at step 52 a test is done to determine whether the string entered in the address field (URL) is recognized. An address is considered recognized if the client has made contact with that URL and thus the list of recognized URLs consists of any past URL the user has been able to access. If the string is recognized (valid) then the client is connected to the target URL, if the string is not recognized (inconsistent with predefined URL) then the process continues to try to correct the string. It is seen that the list of recognized URLs were specified by the user because it is a list of previously contacted URLs, and in order for the URLs to be previously contacted they must have originally been specified by the user in an attempt to access those URLs)

Examiner recites the same rationale to combine used in claim 31.

21. **As to Claim 44,** Nielsen-Berstis discloses the invention as claimed as described in claim 43**, further comprising:**

providing a predefined default target valid FQDN by a FQDN default setter, wherein the FQDN default setter is used by the FQDN mapping module (Figure 5 of Nielsen discloses if the invention is unable to conclusively correct the invalid URL it will return a page to the user with the candidate URL and a request for other candidates. This is seen to be a default target valid FQDN, as it is the default if the correction to the invalid URL is not readily available. This page is seen to be predefined because it has been set by the invention to be the default page)**.**

22. **As to Claim 45,** Nielsen-Berstis discloses the invention as claimed as described in claim 43**, wherein the FQDN mapping module is configured to provide a mapping between the invalid URL request and the target valid FQDN** (Figure 3 of Nielsen discloses a table that holds the invalid URLs and the correct URLs that they have been mapped to and then figure 5 discloses returning the correct URL from the originally invalid URL (545). This is seen to be having provided a mapping between the invalid URL and target valid FQDN)**.**

23. **As to Claim 48,** Nielsen-Berstis discloses the invention as claimed as described in claim 43**, wherein the URL detection module, the URL redirection module, and the FQDN translation module execute in a browser** (Column 5 lines 20-25 of Nielsen disclose the user's computing device running a network browser such as a WWW browser software. Then column

2 lines 55-60 disclose the spell checking will transparently correct the URL and instruct the browser to return the document addressed by the corrected URL. Since the spell checker is able to instruct the browser it is seen to be executing inside the browser. As such it is seen that all associated modules are executing within the browser).

24. Claims 34, 35, 40, 41, 46 and 47 are rejected under 35 U.S.C. 103(a) as being unpatentable over Nielsen-Berstis and further in view of US Pat. 6151624 to Teare et al. (hereinafter "Teare").

25. **As to Claim 34,** Nielsen-Berstis discloses the invention as claimed as described in claim 31. Nielsen-Berstis does not explicitly disclose **wherein the URL request comprises an alias, wherein the alias is stored in the FQDN mapping module.**

However, Teare discloses this (Figure 6 of Teare discloses receiving a real name entry in a browser's network address field (602) and then looking up the real name in an override table (606). The override table is shown in figure 10 to map addresses to specific URLs)

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the system of claim 31 as disclosed by Nielsen-Berstis, with having the URL request comprise an alias and having the alias be stored in the mapping module disclosed by Teare. One of ordinary skill in the art would have been motivated to combine because it is desirable to have a way to access information available over the Web using a natural language word or "real" name associated with the information (column 4 lines 4-6 of Teare).

26. **As to Claim 35,** Nielsen-Berstis-Teare discloses the invention as claimed as described in claim 34**, wherein the FQDN mapping module comprises a mapping of the alias to the target valid FQDN** (Figure 6 of Teare discloses receiving a real name entry in a browser's network address field (602) and then looking up the real name in an override table (606). The override table is shown in figure 10 to map addresses to specific URLs)**.**

Examiner recites the same rationale to combine used in claim 34.

27. **As to Claim 40,** Nielsen-Berstis discloses the invention as claimed as described in claim 37. Nielsen-Berstis does not explicitly disclose **wherein the URL request comprises an alias, wherein the alias is stored in the FQDN mapping module.**

However, Teare discloses this (Figure 6 of Teare discloses receiving a real name entry in a browser's network address field (602) and then looking up the real name in an override table (606). The override table is shown in figure 10 to map addresses to specific URLs)

Examiner recites the same rationale to combine used in claim 34.

28. **As to Claim 41,** Nielsen-Berstis-Teare discloses the invention as claimed as described in claim 40**, wherein the FQDN mapping module comprises a mapping of the alias to the target valid FQDN** (Figure 6 of Teare discloses receiving a real name entry in a browser's network address field (602) and then looking up the real name in an override table (606). The override table is shown in figure 10 to map addresses to specific URLs)**.**

Examiner recites the same rationale to combine used in claim 34.

29. **As to Claim 46,** Nielsen-Berstis discloses the invention as claimed as described in claim 43. Nielsen-Berstis does not explicitly disclose **wherein the URL request comprises an alias, wherein the alias is stored in the FQDN mapping module.**

However, Teare discloses this (Figure 6 of Teare discloses receiving a real name entry in a browser's network address field (602) and then looking up the real name in an override table (606). The override table is shown in figure 10 to map addresses to specific URLs)

Examiner recites the same rationale to combine used in claim 34.

30. **As to Claim 47,** Nielsen-Berstis-Teare discloses the invention as claimed as described in claim 46**, wherein the FQDN mapping module comprises a mapping of the alias URL request to the target valid FQDN** (Figure 6 of Teare discloses receiving a real name entry in a browser's network address field (602) and then looking up the real name in an override table (606). The override table is shown in figure 10 to map addresses to specific URLs)**.**

Examiner recites the same rationale to combine used in claim 34.

Conclusion

31. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after

the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to KEVIN S. MAI whose telephone number is (571)270-5001. The examiner can normally be reached on Monday through Friday 7:30 - 5:00 EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Bunjob Jaroenchonwanit can be reached on 571-272-3913. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/K. S. M./
Examiner, Art Unit 2456

/Jeffrey Pwu/
Supervisory Patent Examiner, Art Unit 2446